USF UNIVERSITY OF SAN FRANCISCO

CHANGE THE WORLD FROM HERE

Inheritance

Casting

Chris Bryan ◆ cbryan2@usfca.edu
Department of Computer Science

Casting

- **Upcasting** (or implicit casting)
  - References an object as its superclass
  - Only able to access methods in direct or indirect superclasses
- **Downcasting** (or explicit casting)
  - References an object as its subclass
  - Allows access to methods in the subclass

Casting

- Does not change the *type* of object, only changes the *reference* to an object
- With *overridden* methods, will call the method associated with the object type (not the reference)
- Can use casting to create *generalized* methods that work on multiple subclasses

Casting Syntax

java.lang.Object
    java.lang.Number
        java.lang.Double

All Implemented Interfaces:

Serializable, Comparable<Double>

```
public final class Double
extends Number
implements Comparable<Double>
```

The `Double` class wraps a value of the primitive type `double` in an object. An object of type `Double` contains a single field whose type is `double`.

In addition, this class provides several methods for converting a `double` to a `String` and a `String` to a `double`, as well as other constants and methods useful when dealing with a `double`.

Since:

JDK1.0

https://docs.oracle.com/javase/8/docs/api/java/lang/Double.html

Casting Syntax

```
1 // Upcasting Examples
2 Number n = new Double(3.14);
3 Object o = n;
4
5 // Downcasting Example
6 Double d = (Double) n;
```


Chris Bryan ◆ cbryan2@usfca.edu
Department of Computer Science
UNIVERSITY OF SAN FRANCISCO

Explicit Casting

```
1 // Throws a ClassCastException
2 Object a = new String("3.14");
3 Double b = (Double) a;
```

https://docs.oracle.com/javase/8/docs/api/java/lang/ClassCastException.html

Chris Bryan ◆ cbryan2@usfca.edu
Department of Computer Science
UNIVERSITY OF SAN FRANCISCO

CHANGE THE WORLD FROM HERE

Chris Bryan ◆ cbryan2@usfca.edu
Department of Computer Science